

Petitioner's Demonstratives

Google LLC
v.
Sandpiper CDN LLC.

IPR2025-00826
U.S. Patent No. 9,021,112

August 6, 2026

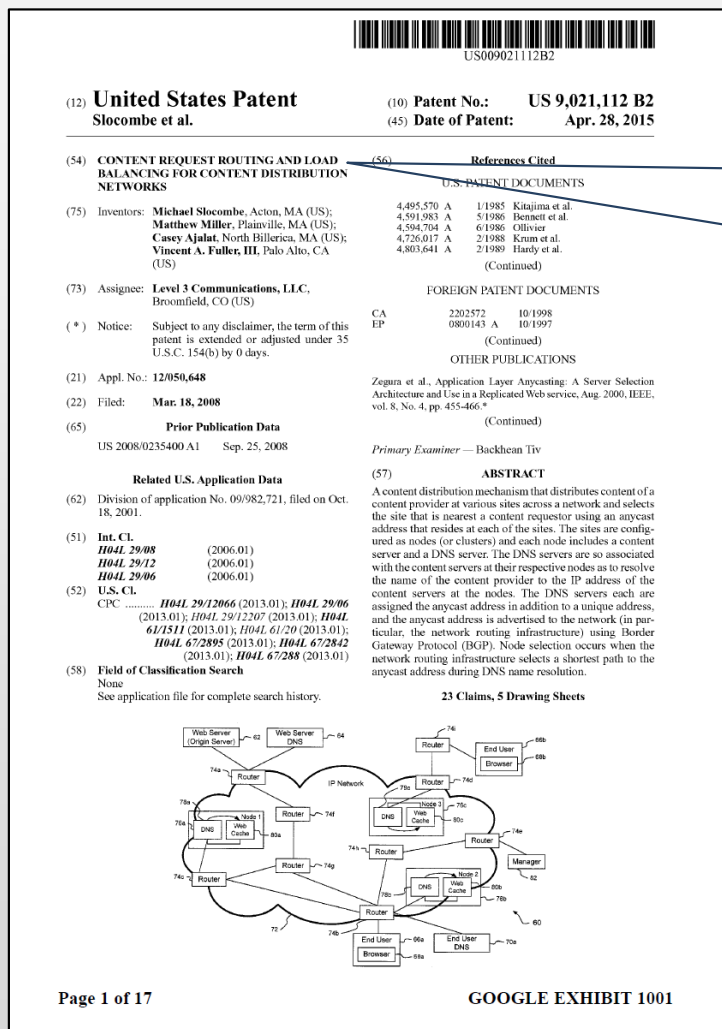
Overview – '112 Patent

CONTENT REQUEST ROUTING AND LOAD BALANCING FOR CONTENT DISTRIBUTION NETWORKS

Petitioner challenges all claims (1-23) based on obviousness.

“A” grounds based on *Glines*

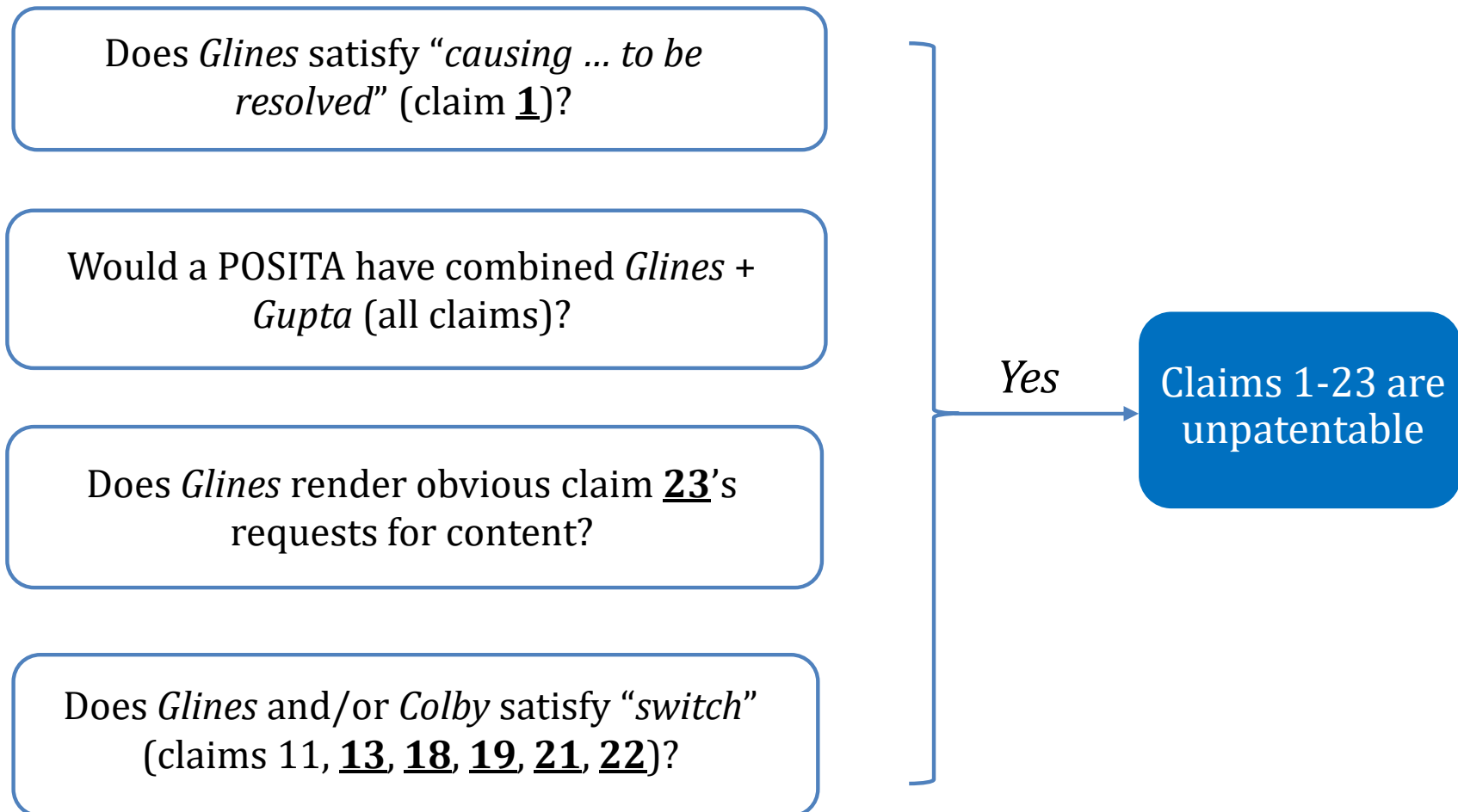
“B” grounds based on *Glines + Gupta*



Ex. 1001 ("112 patent") at Cover.

Disputed Issues

PO raises several disputes across Petitioner's grounds:



Glines satisfies:

“causing said hostname to be resolved ... by an Internet Service Provider (ISP) DNS server”

Claim 1

Glines Satisfies “causing ... to be resolved”

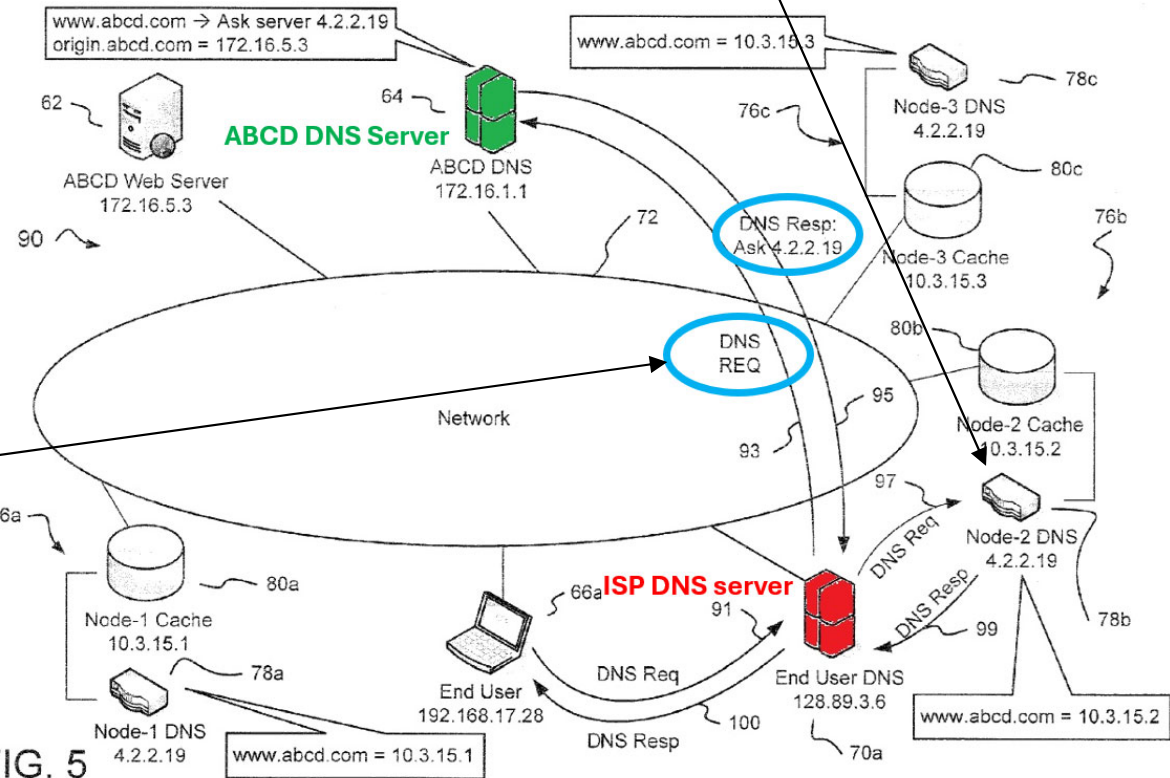
- (c1) causing said hostname to be resolved to said common anycast address by an Internet Service Provider (ISP) DNS server; and then
- (c2) by one of said plurality of CDN DNS servers, resolving said hostname to identify an IP address for use by the client to retrieve the content from a content server.

(c2) “resolving” by the CND DNS server to IP address

*(c1) “causing” resolution to anycast address by **DNS server 70a** querying **ABCD DNS server 64***

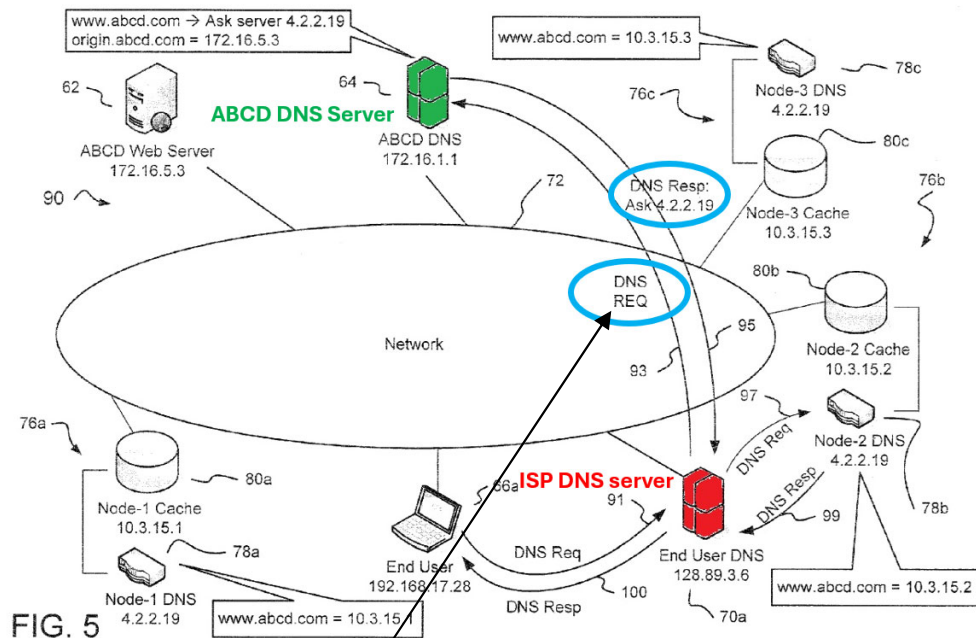
'112 patent at Fig. 5.

FIG. 5



Glines Satisfies “causing ... to be resolved”

'112 patent

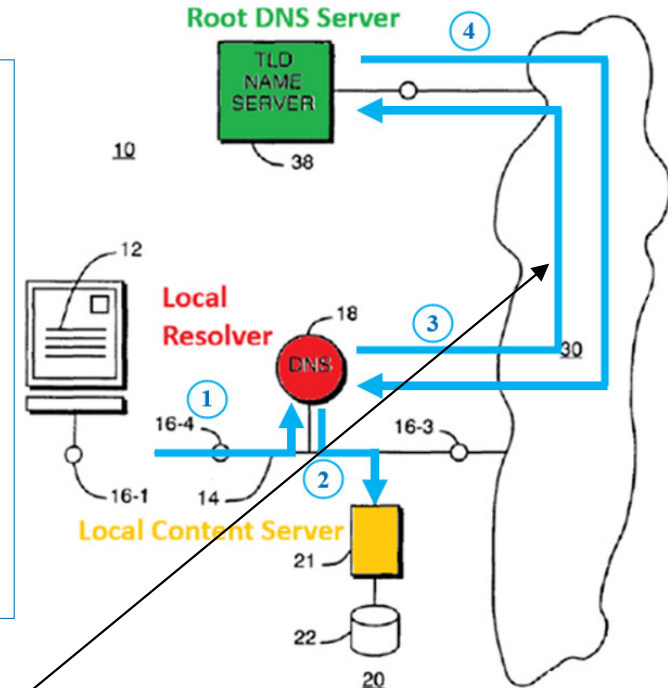


'112 patent at Fig. 5 (annotated).

(c1) “causing” resolution by **DNS server**
70a querying **ABCD DNS server 64**

Glines

1. User sends a request for content to local DNS resolver 18.
2. Local DNS resolver 18 checks for local content.
3. Local DNS resolver 18 requests resolution from root DNS server 38.
4. Anycast address is returned to local DNS resolver 18.



(c1) “causing” resolution by **local**
resolver 18 querying **root DNS server 38**

Glines Satisfies “causing ... to be resolved”

- (c1) causing said hostname to be resolved to said common anycast address by an Internet Service Provider (ISP) DNS server; and then
- (c2) by one of said plurality of CDN DNS servers, resolving said hostname to identify an IP address for use by the client to retrieve the content from a content server.

'112 patent at claim 1.

Patent Owner advances an *incorrect construction* to avoid *Glines*:

Patent Owner

the (ISP) DNS server must “*perform*[] the resolution” or “*resolve*[] ‘said hostname’” itself (POR, 25-26).

UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner

v.
SANDPIPER CDN, L.
Patent Owner

Case IPRO2015-0002
U.S. Patent No. 9,021,____

PATENT OWNER RESPONSE
UNDER 37 C.F.R. § 42.120

Mail Stop “PATENT BOARD”
Patent Trial and Appeal Board
U.S. Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22311-1450

POR 24-27.

Pet. 19-23; Pet. Reply 2-11; POR 24-27; PO Sur-Reply 1-10.

Glines Satisfies “causing ... to be resolved”

The specification discloses (c1)'s anycast resolution *by query*, like *Glines*:

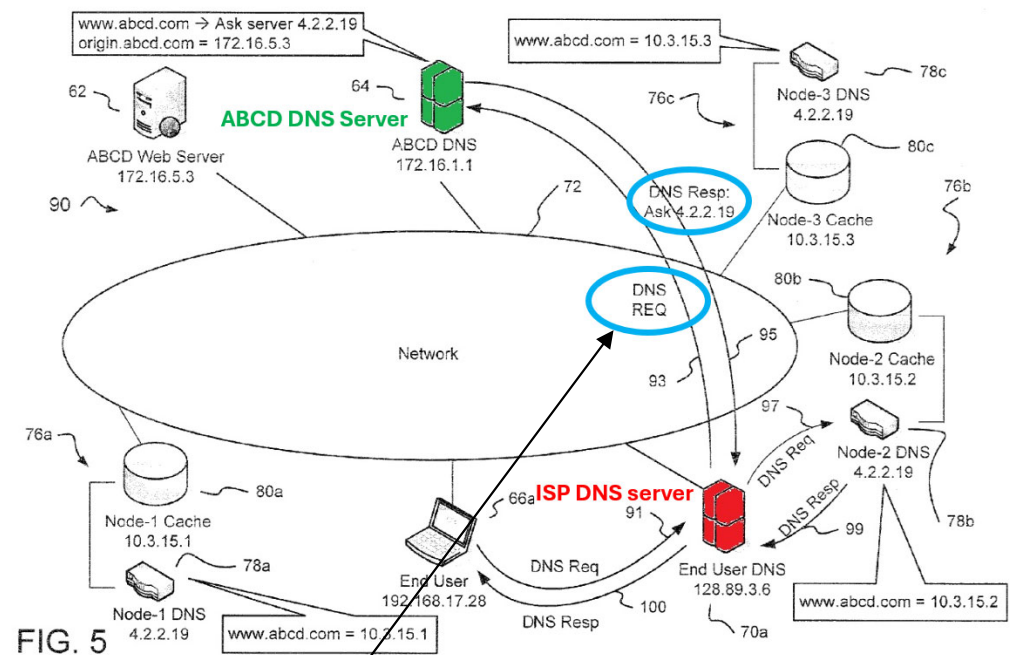
'112 Patent

That is, the end-user station (acting as a content requestor) sends a request to the local DNS server 70a (“DNS Req 92”). That server 70a resolves the name, ultimately by sending a request to the CDN DNS anycast address. The DNS server 70a sends a request to the ABCD DNS server 64 (“DNS Req 94”), which returns the anycast address of the CDN nodes 76 to the requesting DNS server 70a (DNS Resp 96). The DNS server 70a then transmits a DNS name resolu-

'112 patent at 7:2-9 (citing Fig. 5).

The end-user DNS systems, e.g., end user DNS system 70a, resolve to the common address. That is, the end-user DNS system 70a knows which DNS system (in this example, the DNS system 64) has an address for a high level domain server, e.g., .com, org, and maintains tables of all domain names and knows which server (authoritative DNS server) to consult for the address of the domain server. Thus, the address lookup table in the DNS system 64 is configured to indicate that a server corresponding to the common address can resolve the domain name of the content provider to an IP address.

'112 patent at 5:18-31 (citing Fig. 3).



'112 patent at Fig. 5.

(c1) "causing" resolution by **DNS server 70a**
querying **ABCD DNS server 64**

Glines Satisfies “causing ... to be resolved”

Admitted prior art discloses *query-based* DNS resolution by ISP DNS servers:

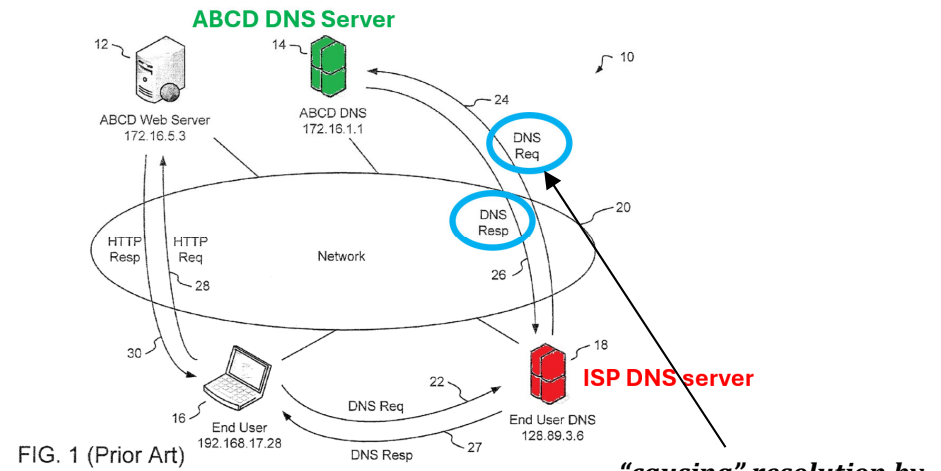
'112 Patent

The end-user system 16 requests a DNS resolution for the host name (“www.abcd.com”) from the end-user DNS system 18 (“DNS Req 22”). The end-user DNS system 18 determines which of the DNS systems that make up the DNS for the network 20 can resolve this host name by sending a DNS request to the authoritative DNS system 14 (“DNS Req 24”). The authoritative DNS system 14 resolves the name to an IP address and returns a response containing the IP address to the end-user 16 system via the end-user DNS system 18 (“DNS Resp 26”).

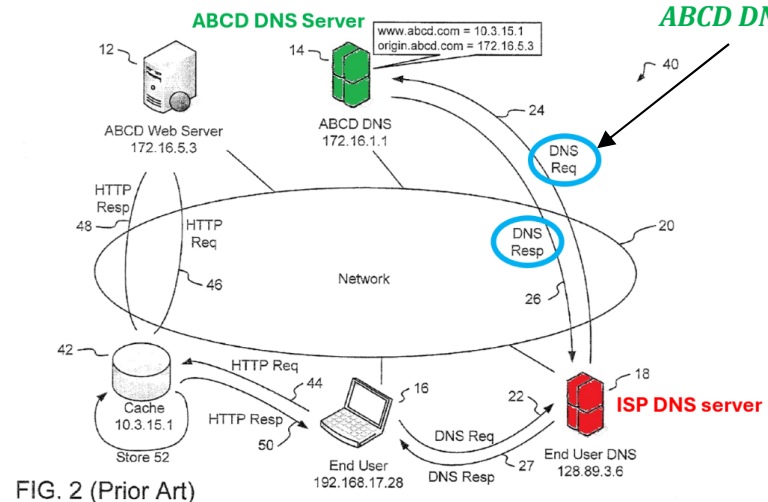
'112 patent at 3:13-54 (citing Fig. 1).

FIG. 2 illustrates a Web transaction with reverse proxy caching 40. That is, the transaction 40 is the same basic Web transaction as shown in FIG. 1, but now employs a reverse proxy content server (shown as a cache server) 42 acting on behalf of the Web server 12. During reverse-proxy caching,

'112 patent at 3:55-4:17 (citing Fig. 2).



“causing” resolution by
DNS server querying
ABCD DNS server



'112 patent at Figs. 1, 2.

Pet. 19-23; Mowry ¶¶ 141-163; Pet. Reply, 2-11; Mowry Supp. ¶¶ 5-23; '112 patent at 3:13-4:17, Figs. 1, 2.

Glines Satisfies “causing ... to be resolved”

A POSITA would have understood that “causing” resolution encompasses well-known DNS by query techniques:

Dr. Mowry:

36. DNS (Domain Name System) resolution is the process of translating human-readable domain names, such as “example.com,” into IP addresses (strings of numbers) that computers use to communicate. When a user seeks content from a particular web site, the user may type the website domain name into a web browser. Typically, the browser checks its local cache, followed by the operating system’s cache, to find the corresponding IP address. If the domain name is not cached, the query is sent to a DNS resolver, typically provided by the user’s Internet Service Provider (ISP) or a third-party service. The resolver acts as an intermediary, performing recursive queries by contacting DNS servers in a hierarchical order: root servers, top-level domain (TLD) servers (e.g., .com), and finally the domain’s authoritative DNS server (e.g., the “example” company server, often referred to as the origin server), which maintains the actual IP address of the requested domain name.

Mowry ¶¶ 36.

Local DNS server querying other DNS servers to resolve content

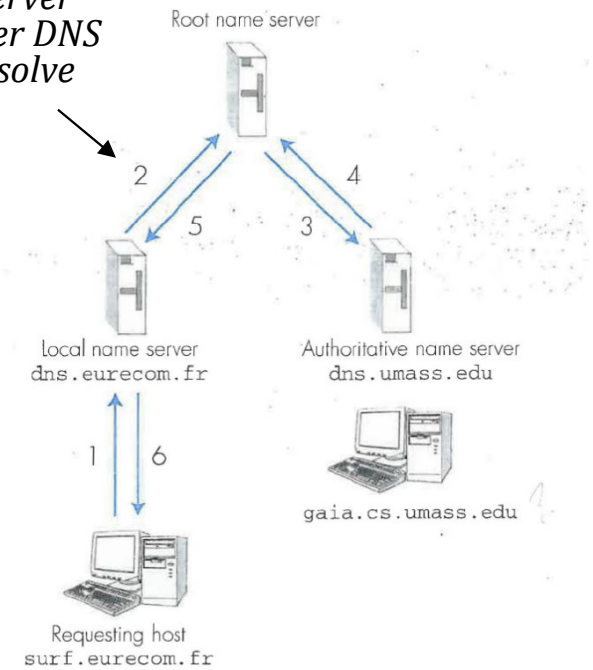


Figure 2.19 + Recursive queries to obtain the mapping for gaia.cs.umass.edu

EX-1011 (“Kurose”) at Fig. 2.19 (illustrating recursive queries).

Glines Satisfies “causing ... to be resolved”

Patent Owner is ***wrong*** that *Glines*’s local DNS resolver 18 is just a cache and plays “no role in resolving a hostname to an anycast address.”

Glines:

typically formulated as a message sent to a local address resolver. If the
“example.com” domain is not located in a local domain, the local resolver will then
5 proceed to consult public name servers that have been defined as being authoritative
for the “example.com” domain. The root domain name servers defined, for

* * *

the browser or local resolver then sends out a DNS request as a UDP datagram to an
anycast message to the group. This will now resolve the IP address of
www.example.com.

Glines at 7:1-13.

“[c]aching can occur at every level in the DNS hierarchy, so ***it was typical*** for the ISP DNS server ***to first check locally*** . . . the ISP DNS server ***would [then] query*** an authoritative DNS server for the requested content.”

Mowry Suppl. ¶ 18; Mowry ¶¶ 36-40.

Glines Satisfies “causing ... to be resolved”

Petitioner’s argument is not “new”:

Glines further teaches that, when a user “associated with an Internet Service Provider” enters a URL (which includes a hostname, *supra* Limitation [1c]) in a web browser, local DNS resolver 18 sends a request to a root DNS server 38. Glines, 13:24-14:12; *also id.*, 11:24-28; EX-1008 ¶¶154-156. In turn, root DNS name server 38 is “programmed to ... return the anycast address 50.100.20.1” of name servers 53-1, 53-2, 53-3 that “have been designated as being the authoritative name servers for ‘example.com.’” Glines at 14:4-15:4; *also id.*, 7:1-7, 9:9-15. The browser or local resolver can then “send[] out a DNS request as a UDP datagram to [the] anycast message” to resolve the content. *Id.*, 7:9-13; *also id.*, 6:16-17.

* * *

Glines’s local DNS resolver, an ISP DNS server, thus causes the hostname entered as part of the URL to be resolved to the name servers’ common anycast address, as claimed. EX-1008 ¶¶154-156.

Pet. 21-22.

Glines Satisfies “causing ... to be resolved”

Glines’s local resolver 18 discloses the “causing ... to be resolved”:

Glines

as a UDP datagram to a name server. In the case where the user is associated with
25 an Internet Service Provider (ISP) operating the local area network 20, this first
name request is made to a local DNS resolver 18, to determine the location of the
domain “example.com”. The DNS resolver 18 determines whether or not the

Glines at 13:21-27.

typically formulated as a message sent to a local address resolver. If the
“example.com” domain is not located in a local domain, the local resolver will then
5 proceed to consult public name servers that have been defined as being authoritative
for the “example.com” domain. The root domain name servers defined, for
example, by the Internic, may be programmed to return the common address of the
group of name server/content server peers that share a common anycast address.

Glines at 7:1-7.

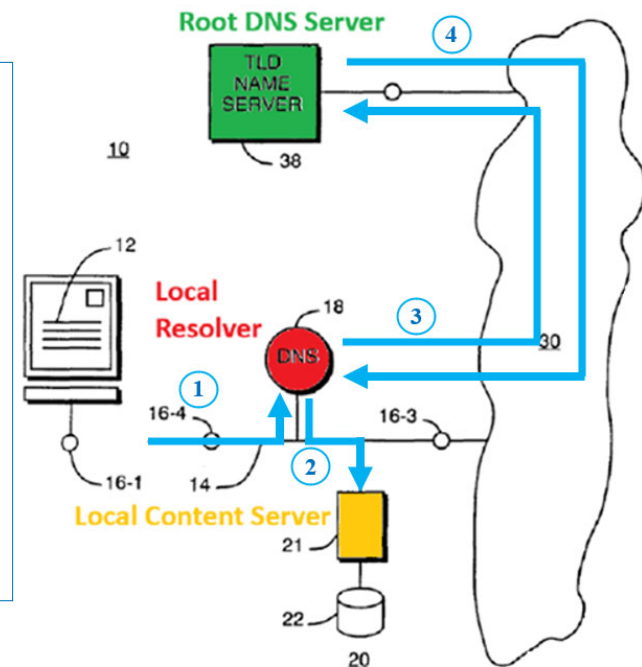
Having failed to resolve the requested name locally, a request to resolve the
location of “example.com” is then sent to a root DNS server 38 in step 108. In this

Glines at 14:8-12.

However, in accordance with the invention, the root DNS name server 38
has been programmed to instead return the anycast address 50.100.20.1.

Glines at 14:17-18.

1. User sends a request for content to local DNS resolver 18.
2. Local DNS resolver 18 checks for local content.
3. Local DNS resolver 18 requests resolution from root DNS server 38.
4. Anycast address is returned to local DNS resolver 18.

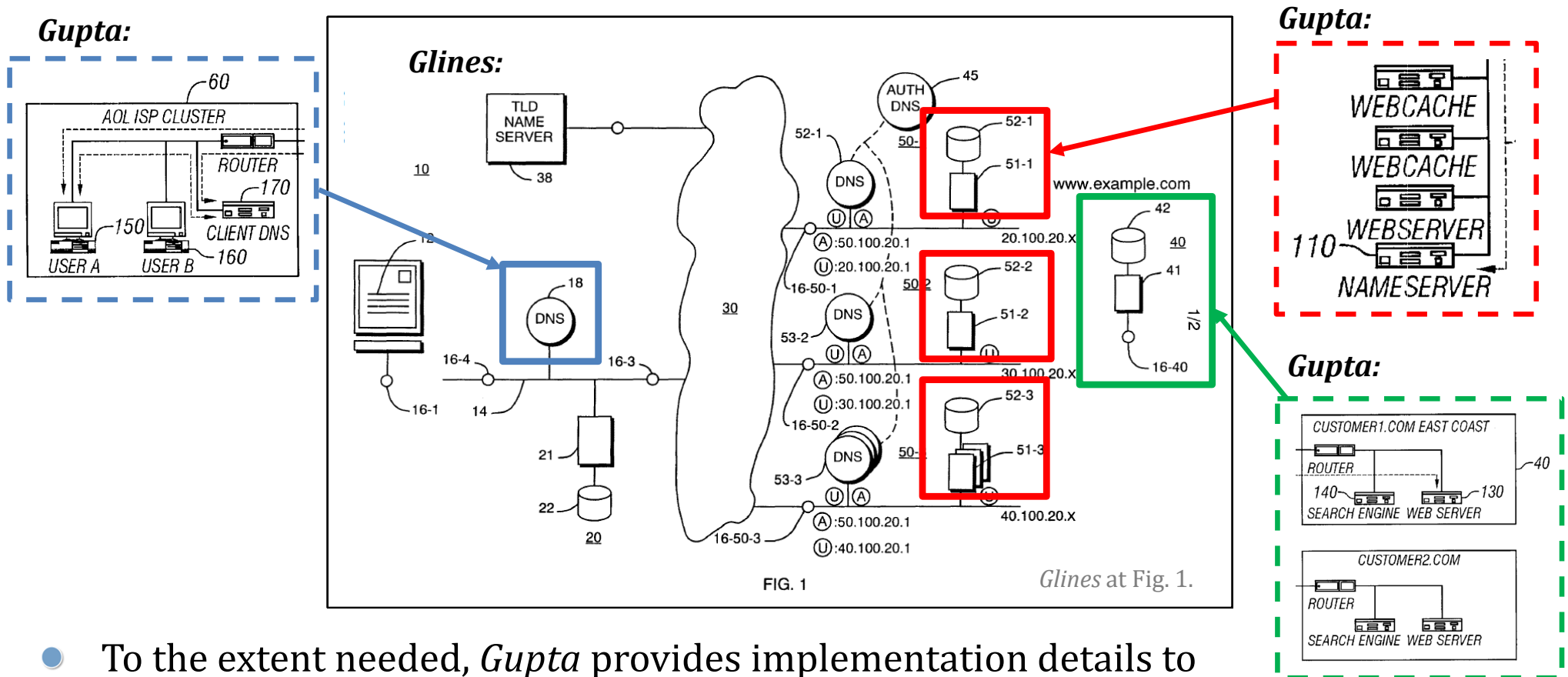


Glines at Fig. 1 (excerpted, annotated).

A POSITA would have combined
Glines + Gupta

“B” Grounds, all claims

A POSITA Would Have Combined *Glines* + *Gupta*



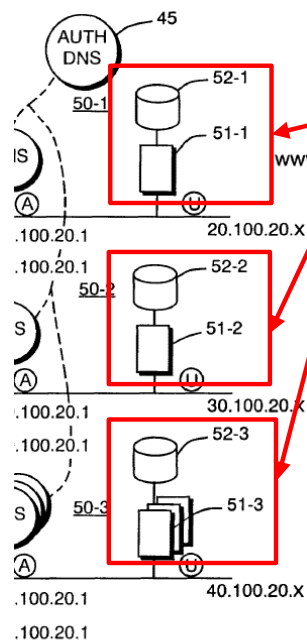
- To the extent needed, *Gupta* provides implementation details to *Glines* that satisfy:

- “(ISP) DNS server” (claims 1, 18, 19, 21, 22, 23)
- “**plurality of content servers**” (claim 1)
- “**plurality of content providers**” (claims 13, 18, 22, and 23).

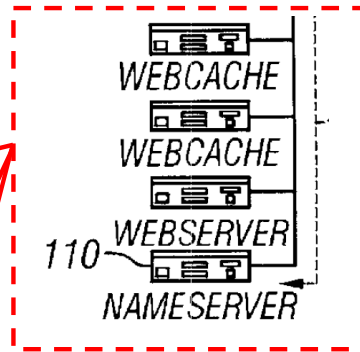
Pet. 14-16, 23-25, 39-41; Mowry ¶¶ 109-118, 164-169, 264-267; Pet. Reply 11-13, 17-18, 24-28.

A POSITA would have implemented *Glines* based on *Gupta's* multiple webcache servers

Glines:



Gupta:



Gupta at Fig. 1.

Glines at Fig. 1.

“a plurality of content servers”
associated with *“each”* CDN DNS server

Petitioner cited *evidence* and explained *why*:

112. As discussed with respect to Glines, employing multiple WebCache servers, as taught by Gupta and suggested by Glines, would have improved the ability of each name server to handle manage high request volumes without degrading performance. For example, each name server would have been able to load balance requests across those servers during periods of high traffic. This also would have improved managing server faults without significant performance degradation. For example, in the event of a content server failure, other associated content servers would handle content requests and ensure uninterrupted service. This also would have ensured a name server remains operational during scheduled maintenance or unexpected outages, and avoided single points of failure.

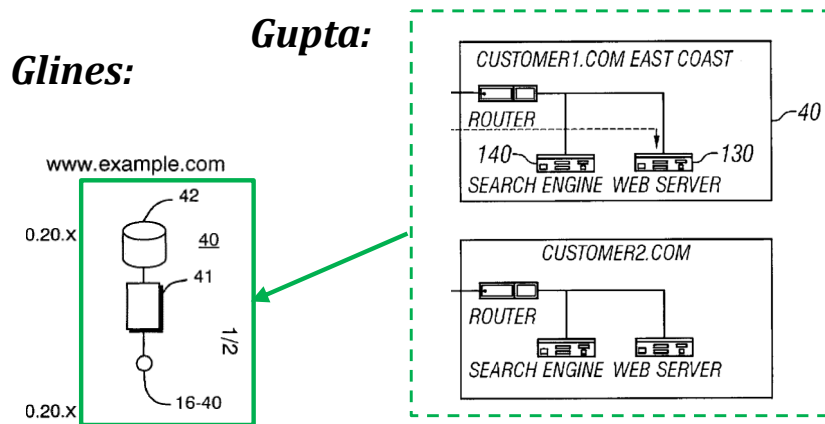
113. Associating multiple content servers with each name server in Glines's system also would have increased Glines's storage capacity and allowed more content to be stored at each name server. This would have increased the ability for users to retrieve content from the nearest name server in Glines's system, thereby increasing system performance by reducing latency.

114. Therefore, a POSITA would have been motivated to combine Glines with Gupta in this manner to improve performance, reliability, and scalability of Glines's network.

Mowry ¶¶ 112-114.

Pet. 7-8, 14-16; Gupta at 4:45-67, 5:16-53, 6:10-21, 10:4-22, Fig. 1; Mowry ¶¶ 68-74, 109-118; DI 21-24.

A POSITA would have implemented *Glines* based on *Gupta*'s multiple "customer" content sources



Glines at Fig. 1.

Gupta at Fig. 1.

"a plurality of content providers"

Petitioner cited *evidence* and explained *why*:

267. It would have been obvious to apply these teachings to Glines's system.

The combined system would have stored and provided content from its content servers from multiple content providers (e.g., multiple customers and their associated websites, etc.). The combination would have been nothing more than combining known elements in the prior art according to understood principles, with an expectation of success. Doing so would have been expected and well within the capabilities of a POSITA to, for example, maximize the usefulness and value of Glines's system by providing Glines's service to more than one customer/content provider.

Mowry ¶ 267.

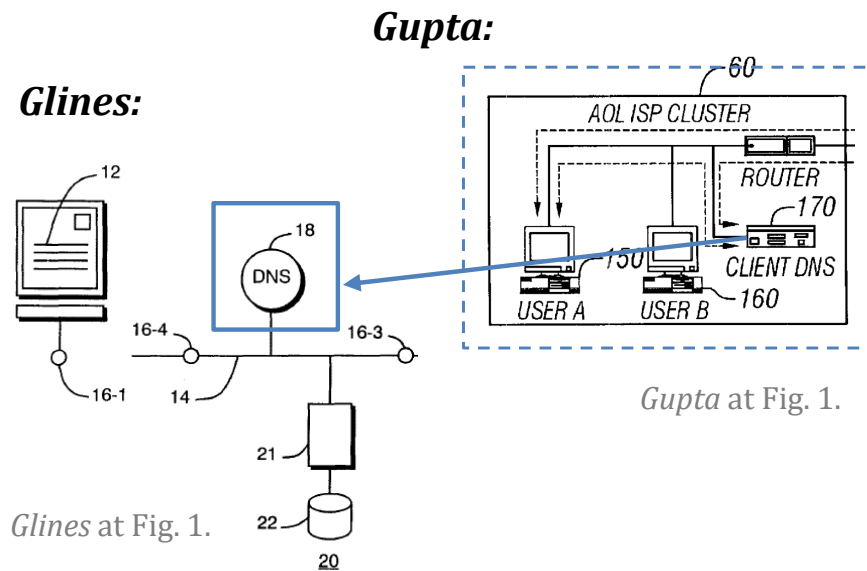
"[t]his asserted advantage is supported in the record, and by Petitioner's reasoned explanation."

DI 23.

Pet. 40-41; Gupta at 7:8-20, 8:53-59, Fig. 1; Mowry ¶¶ 264-267.

A POSITA would have implemented *Glines* based on *Gupta's* ISP 60/client DNS server 170

Petitioner cited **evidence** and explained **why**:



*“Internet Service Provider
(ISP) DNS server”*

Combination would have “*caus[ed] ... resolution*” to anycast address based on *Glines’s* teachings.

167. Gupta’s ISP 60 thus performs the same functions as Glines’s local DNS resolver 18 (e.g., resolving a host name into an IP address). In view of Glines’s teaching that local DNS resolver 18 is used “where the user is associated with an Internet Service Provider (ISP),” Glines, 13:24-27, a POSITA would have understood Gupta’s ISP 60 to be an obvious and known substitute for Glines’s local resolver 18.

168. A POSITA would have been motivated to make the modification to ensure compatibility with “typical” internet service architecture, *e.g.*, Gupta, 4:45-51, 5:65-6:9, and to achieve the speed and efficiency associated with ISP-provided DNS servers, *see, e.g.*, EX-1011, 128; EX-1013, 5:35-54.

169. Given the ubiquity of ISP-provided DNS servers and the disclosures of Gupta and Glines, a POSITA would have viewed the combination as nothing more than the simple substitution of known elements in the prior art according to understood principles to yield predictable results. A POSITA also would have had an expectation of success in the substitution. Combining Glines with Gupta in this way would not alter the features of Glines discussed elsewhere in this declaration.

Mowry ¶¶ 167-169.

A POSITA Would Have Combined *Glines* + *Gupta*

- Petition articulated reasons to combine each feature.
- *Glines* and *Gupta* are ***not*** “incompatible” or “vastly disparate.”
- Use of anycast addressing ***remains*** in the combination.



“Where, as here, the grounds for obviousness are based on a specific combination of references, arguments that attack the disclosures of the two references individually lack merit.”

Palo Alto Networks, Inc. v. Centripetal Networks, LLC, 122 F.4th 1378, 1386 (Fed. Cir. 2024)
(citation and internal quotes omitted).

Pet. Reply at 11-13, 24-28; POR 27-29, 47-55; PO Sur-Reply at 4, 9-10, 23-26.

A POSITA Would Have Combined *Glines* + *Gupta*

- PO's "hindsight" argument is limited to the "(ISP) DNS server," which *Glines* teaches.
- Regardless, no "hindsight" in combination with *Gupta*.

Applicant: *McCanne* did not perform the initial "causing" resolution by an ISP DNS server:

As indicated in the paragraph above, the APAR routing system of *McCanne* utilizes a specially designed CDN DNS server situated within an ISP that is configured with an anycast address per each associated CDN. *McCanne* does not, however, describe the initial step of using an ISP DNS server to resolve the request to the anycast address as is recited in claim 1 step (d1). For at least these reasons, *McCanne* does not teach each and every limitation of claim 1. As

'112 prosecution history (EX-1007) at 331.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re the Application of:

Applicant : Michael Slocombe et al.	Confirmation No: 9055
Appln. No. : 12/050,648	Group Art Unit: 2449
Filed : 18 March 2008	Examiner: Ashokkumar B. Patel
Title : CONTENT REQUEST ROUTING AND LOAD BALANCING FOR CONTENT DISTRIBUTION NETWORK	Docket No. 0023-US-D1

AMENDMENT AND RESPONSE TO NON-FINAL OFFICE ACTION

Via EFS-Web
MAIL STOP AMENDMENT
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

In response to the Non-Final Office Action mailed 9 June 2009 on the above-referenced application, please amend the above-identified application as follows:

Extension of Time Request is found on page 2 of this paper.

A listing of the claims is found on page 3 of this paper.

Remarks/Arguments begin on page 9 of this paper.

Glines renders obvious:

“first” and “second” requests for content

Claim 23

Independent Claim 23

23. A method comprising:

(A) providing a plurality of Domain Name System (DNS) servers, said plurality of DNS servers sharing a common anycast address, wherein each DNS server is associated with a respective plurality of content servers, wherein each DNS server and respective plurality of content servers is configured to support content delivery for a plurality of content providers;

(B) responsive to a request from a first content provider, said first content provider having a first hostname associated with said first content provider, said first hostname associated with the first content provider, by a first content server associated with said first DNS server; and

(C) responsive to a request from a second content provider, said second content provider having a second hostname associated with the second content provider, by a second content server associated with said second DNS server.

Claim 23:

- two “requests” for content
- content “associated with” two “content providers”
- both requests include the “first hostname”

(B) responsive to a first request for first content associated with a first content provider of said plurality of content providers, said first request including at least a first hostname associated with the first content provider, by a first

(C) responsive to a second request for second content associated with a second content provider of said plurality of content providers, said second request including at least said first hostname associated with the first content pro-

Glines Renders Obvious Claim 23's Two Requests

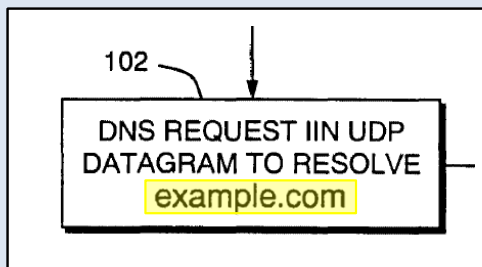
A POSITA would have understood *Glines* renders this use-case obvious:

The following sequence of events may occur with a process implemented according to the invention. First, a user indicates the location of a content file such as by specifying a Uniform Resource Locator (URL) to a browser program, or by clicking on a hyperlink in a displayed document which has an embedded URL. Such a URL may, for example, be “http://www.example.com/homepage.html”.

Glines at 6:13-27.

For example, as the user types a Uniform Resource Locator (URL) into a web browser and/or uses a pointing device, such as a mouse, to click on a hyperlink embedded in a previously viewed web page, the URL for a particular web site is specified to the browser 12.

Glines at 9:16-21.



Glines at Fig. 2.

Glines

“nothing more than applying the art in understood ways to resolve ***particular*** content” and “would have required ***no modification*** of Glines or Gupta to implement”

Mowry ¶¶ 451, 464.

Pet. 54-47; Pet. Reply, 18-22; *Glines*, 6:23-7:24, 13:15-15:10, Figs. 1, 2; Mowry ¶¶ 437-474.

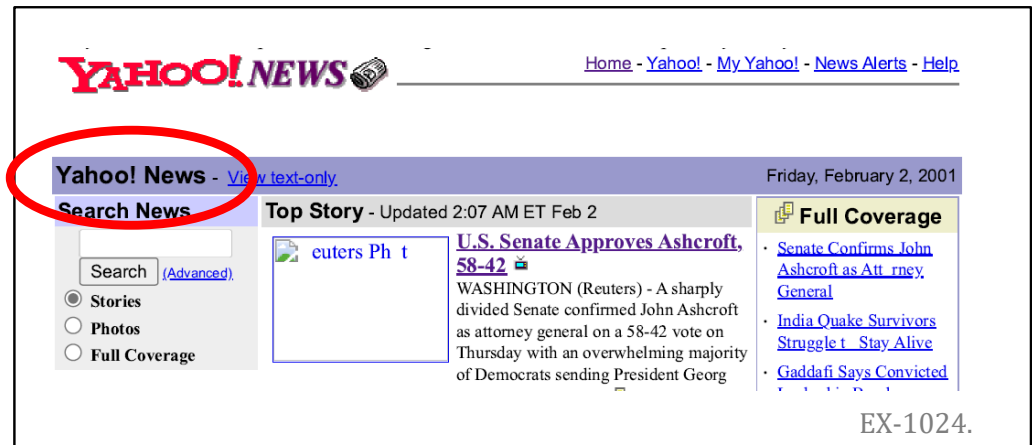
Glines Renders Obvious Claim 23's Two Requests

“first request”:

content “associated with”
Yahoo! News:

includes “first hostname”:

<http://dailynews.yahoo.com>:

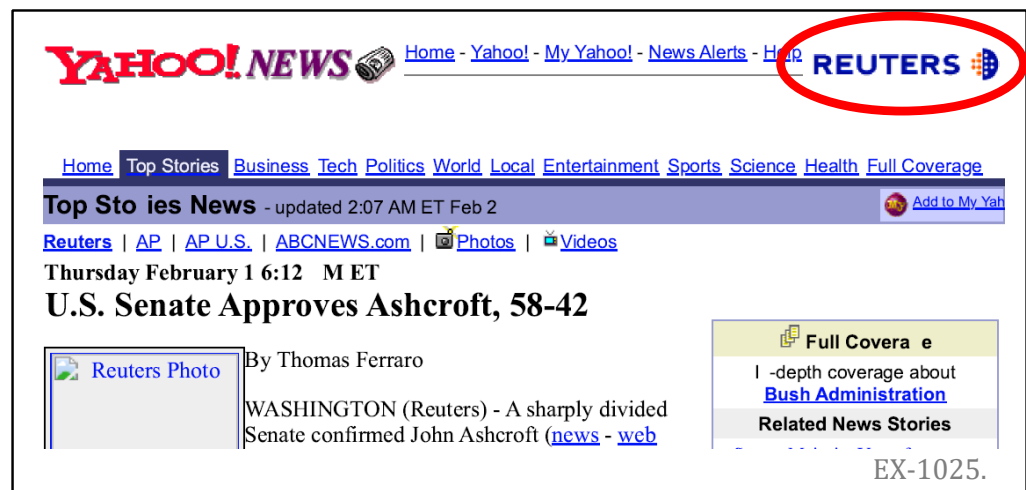


“second request”:

content “associated with”
Reuters:

includes “first hostname”:

http://dailynews.yahoo.com/h/nm/20010201/ts/congress_Ashcroft_dc_71.html



Glines Renders Obvious Claim 23's Two Requests

Evidence supports a POSITA's understanding in view of *Glines*:

449. News aggregator websites, including Yahoo! News, were known in the art before October 18, 2001. *See id.* These websites, such as Yahoo! News, were known to provide links to articles from different news organizations via one webpage for users to browse. *See id.* The links typically would have included a hostname and a path to a particular news source.

Mowry ¶ 449 (citing Ex-1024; EX-1025).

30. As was known, Yahoo! News was launched in 1996 as an “aggregated news page” and by 2001 was in widespread use. EX-1042 (Yahoo! company timeline stating that in August 1996 “Yahoo! News is birthed by aggregated news page created by Yahoo! co-founder”); EX-1046 (NY Times discussing Yahoo! News). Yahoo! News “cull[ed] material from news organizations in exchange for a portion of Yahoo’s ad revenues” and “include[d] hyperlinking to external information sources.” EX-1046 at 3; EX-1043 at 122.

31. In addition to its widespread use, Yahoo! News was also described in Yahoo! patents before the '112 patent. For example, Nazem (EX-1044) discloses a customizable Yahoo! front page that includes links to major news stories.

Mowry Supp. ¶¶ 30-31.

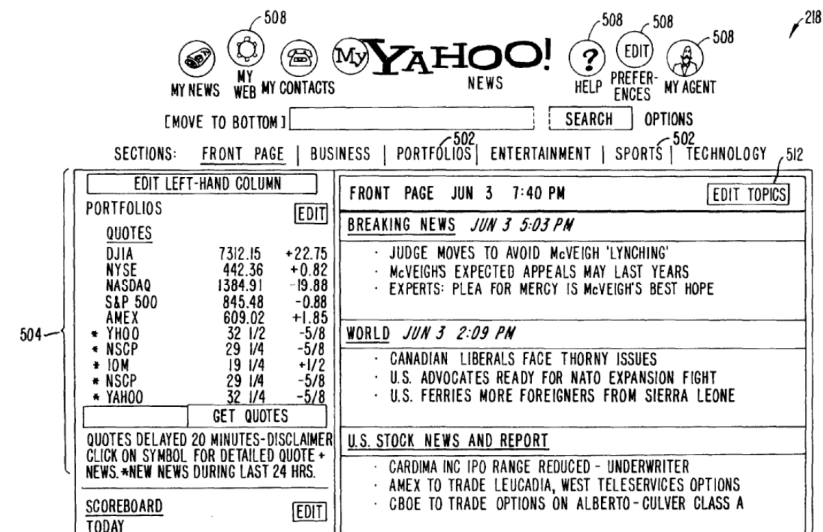


FIG. 5A.

EX-1044 (“Nazem”) at Fig. 5A.

Pet. 54-58 (citing EX-1024; EX-1025); Pet. Reply, 18-22; *Glines*, 11:24-28, 13:24-14:12, Figs. 1, 2; Mowry ¶¶ 437-474; Mowry Suppl. ¶¶ 29-34 (citing EX-1042; EX-1046; EX-1044).

Glines Renders Obvious Claim 23's Two Requests

No missing element in Petition, no improper use of expert testimony:

General knowledge may still be used in an IPR to support a motivation to combine or to demonstrate the knowledge of a person having ordinary skill in the art. *See, e.g., Unification Techs. LLC v. Micron Tech. Inc.*, 2024 WL 3738401, at *7 (Fed. Cir. Aug. 9, 2024) (affirming use of general knowledge in an IPR proceeding in the form of expert testimony and AAPA to demonstrate how a skilled artisan would have understood a prior art reference's disclosure as teaching a claim limitation).

Director's July 31, 2025 Memorandum (EX2023 at 1-2).



Any differences between the claim and prior art amounts to nothing more than “the predictable use of prior art elements according to their established functions.”

KSR Int'l Co. v. Teleflex Inc., 550 U.S. 398, 401 (2007).

Pet. 54-58; Pet. Reply, 18-22; *Glines*, 11:24-28, 13:24-14:12, Figs. 1, 2; Mowry ¶¶ 437-474; Mowry Suppl. ¶¶ 29-34.

Glines and/or Colby satisfies:

“at least one switch”

Claims 11, 13, 18, 19, 21, & 22

Glines Discloses the “at least one switch”

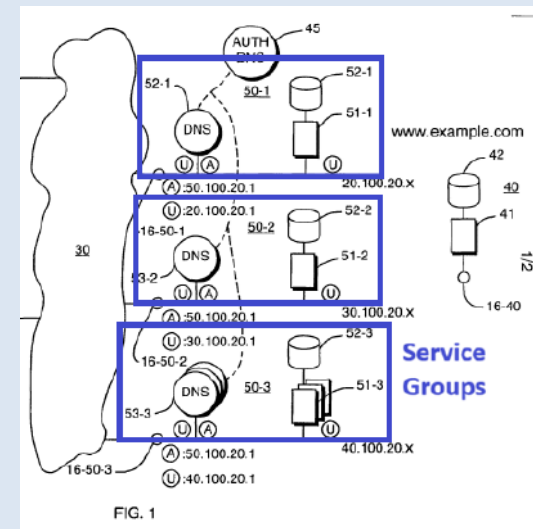
11. The method of claim 1 further comprising:
(c3) by **at least one switch**, selecting the content server from said respective plurality of content servers associated with said resolving CDN DNS.

Glines’s layer 4 switch is a “switch” that “selects” a content server:

* overload

Requests get delivered to the DNS server at the anycast address purely by the topological closeness of the requesting clients.

However, **within the service group, standard load balancing** and replication techniques **can also apply, such as** multiple content servers (returning multiple local IP addresses to a name lookup), **layer 4 switches**, etc. Multiple DNS servers within the group all listening to the anycast address would also be possible.



Glines at 18:20-19:22, Fig. 1.

Glines Discloses the “at least one switch”

Glines teaches **alternative** methods for load balancing, including one using a layer 4 switch:

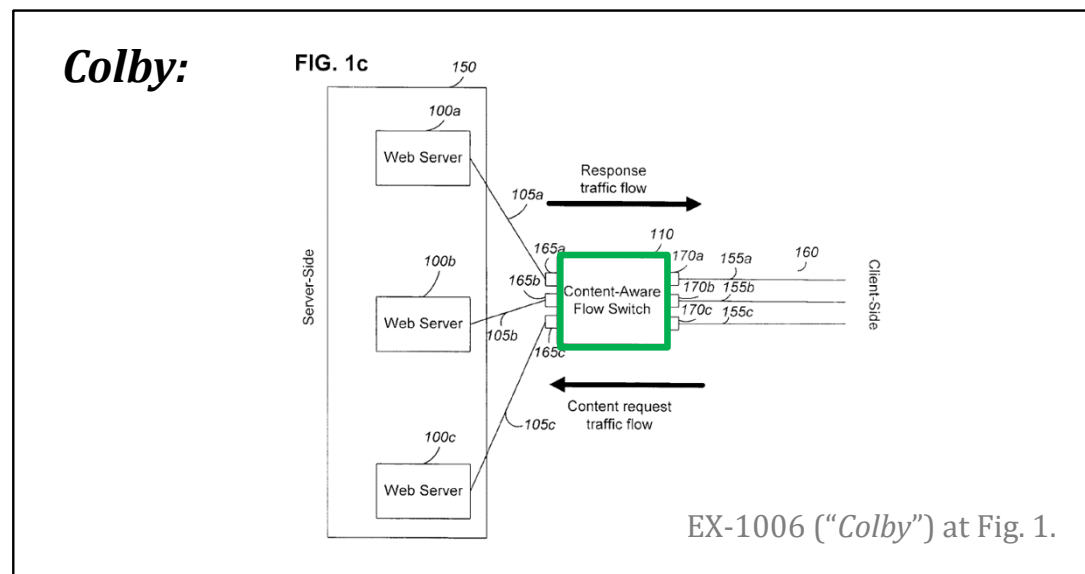
Patent Owner argues that such load balancing is accomplished by returning multiple local IP addresses to a name lookup. Prelim. Resp. 27 (citing Ex. 1002, 19:13–16). However, *Glines* describes this as one of two alternative methods of load balancing, with the other being “layer 4 switches.” Ex. 1002, 19:13–16. Thus, we determine that Petitioner has persuasively shown that *Glines* describes layer 4 switches as a method for selecting a content server within a service group.

Therefore, Petitioner has shown *Glines* to teach limitation [19f]. We have reviewed Petitioner’s additional *Glines*-based assertions for claims 19–20, and determine that not all are supported by the current record.

DI 17.

Colby Discloses the “at least one switch”

Patent Owner does not dispute *Colby*’s “content-aware flow switch” discloses the claimed switch:



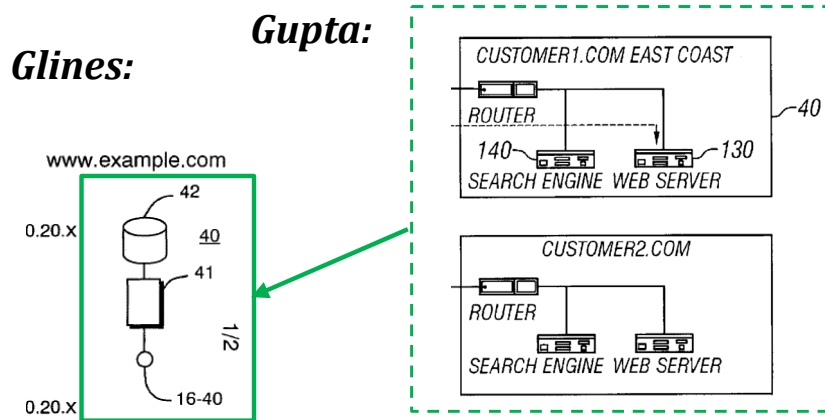
Patent Owner only asserts that *Glines*’s load balancing techniques “do not require the use of an additional switch.”

PO Sur-Reply 46.

Glines & Glines-Gupta satisfy:
“a plurality of content providers”
Claims 13, 18, 22, and 23

Glines-Gupta Satisfies “a plurality of content providers”

Gupta’s “customers” discloses “a plurality of content providers.”



Glines at Fig. 1.

Gupta at Fig. 1.

Gupta:

In the embodiment illustrated in FIG. 1, it should be noted that the POP server network may simultaneously provide traffic management, as well as content hosting to more than one specific customer. For example, customer1 and customer2 share the POP server network. In one example, POP server network may provide traffic load management by directing traffic to either customer1 web server 40 or customer1 web server 45. At the same time, POP server network may be used to host static content for customer2. Still further

Gupta at 7:8-20.

Next a user, such as user 150 enters the name of a customer's web site, step 410. In one embodiment, a user can type-in the name of a customer's web site via keyboard, and in another embodiment, the user can select a hyperlink to the customer's web site from a page the user is viewing, or the like. In the example, the customer's web site is "customer1.com."

Gupta at 8:53-59

Patent Owner ***does not dispute*** the teachings of Gupta.

Glines Satisfies “a plurality of content providers”

Glines alone additionally renders this feature obvious:

[13p]:

13. A method of content delivery for delivering content on behalf of a plurality of content providers, the method comprising:

Petition:

Glines’s “www.example.com” renders obvious delivering content on behalf of one or more content providers.

Adding origin servers and content providers to *Glines*’s system would have been obvious.

Glines Satisfies “a plurality of content providers”

Glines’s “www.example.com” renders obvious delivering content on behalf of one or more content providers

Glines:

file replicas via unicast addressing. Each name server also advertises itself as being authoritative for the domains associated with the content files stored (or to be served from) the associated content server.

* * *

The following sequence of events may occur with a process implemented according to the invention. First, a user indicates the location of a content file such as by specifying a Uniform Resource Locator (URL) to a browser program, or by clicking on a hyperlink in a displayed document which has an embedded URL. Such a URL may, for example, be “http://www.example.com/homepage.html”.

Glines at 6:13-27.

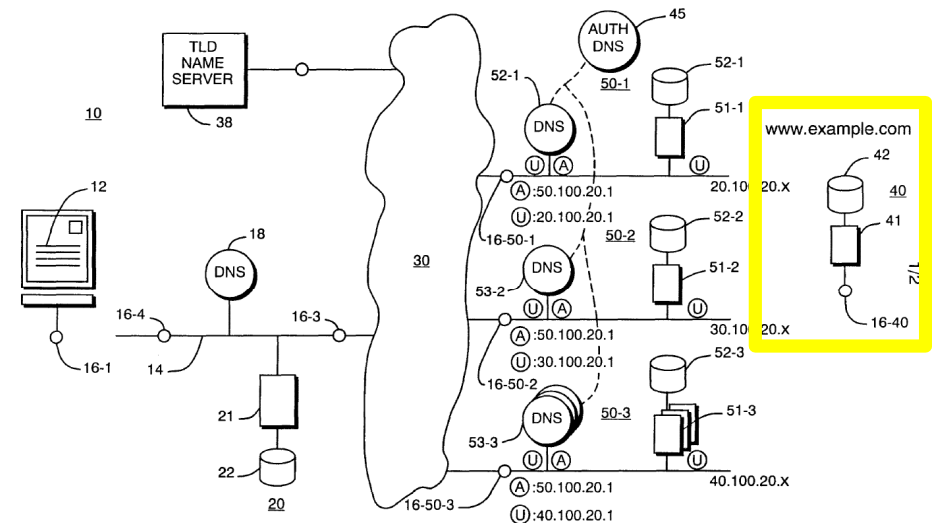


FIG. 1

Glines at Fig. 1.

Glines Satisfies “a plurality of content providers”

- Nothing more than *replicating* origin server 40 and storing content in content servers 50, as *Glines* already teaches.
- Expected results: “enabl[ing] *Glines*’s system to support more (and different) content to fulfill user requests.”



“The missing claim limitation ... involves repetition of an existing element ... until success is achieved.”

B/E Aerospace, Inc. v. C&D Zodiac, Inc., 962 F.3d 1373, 1380-81 (Fed. Cir. 2020).

Gupta and/or Chiou satisfies:

“valid version of the content”

Claims 7, 19

Exemplary Claim 19: “valid version”

19. A method of content delivery, the method comprising:
- (A) providing a plurality of CDN Domain Name System (DNS) servers, said plurality of DNS servers sharing a common anycast address, each DNS server being associated with a respective plurality of content servers, wherein at least some of said plurality of content servers are located at network Points of Presence (POPs);
 - (B) causing said plurality of DNS servers to be authoritative for a hostname associated with a content provider by causing said common anycast address to be associated with said hostname;
 - (C) responsive to a request for content associated with the content provider by a client, said request including at least said,
 - (c1) by one of said plurality of DNS servers, resolving said hostname to identify an IP address for use by the client to retrieve the content from a content server; and
 - (c2) by at least one switch, selecting the content server from said respective plurality of content servers associated with said resolving DNS server,
 - (D) by the content server,

(d1) attempting to serve the content by:

(d11) if a valid version of the content is available on the content server, serving the content; otherwise

(d12) obtaining the content from a content source and then serving the content.

element [19g]

element [19h]

Gupta Satisfies “valid version”

Patent Owner does not dispute *Gupta*’s teachings or the specific reasons to combine with *Glines*.

In this embodiment, the WebCache server software responds to requests for static web content from clients, such as web browsers from memory. In this example, if the requested content does not exist in server memory, the WebCache software will generate a request to a WebServer. When WebServer sends the requested content, the Web-Cache software caches the content in memory.

Gupta at 5:16-22.

“this would have ***improved content distribution*** across *Glines*’s system and improve future service of that content” by “***add[ing] otherwise missing content*** to that content server for future requests.”

Mowry ¶¶ 229-232.

Pet. 33-37, 50-51; *Gupta* at 5:16-40, 13:23-30, Fig. 4; *Mowry* ¶¶ 223-232, 362-386.

Chiou Satisfies “valid version”

Patent Owner does not dispute *Chiou*’s teachings or reasons for combining *Chiou*.

If a cache hit occurs (e.g., the requested data is located in the SSC cache and is valid data), then the read request is forwarded to the cache card that is responsible for the target port in step 86. In step 88, the cache manager on the cache card may retrieve the requested block of data (or files in file-based caching is used) from the predetermined cache location based on a cache location table. In step 90, the cache manager may assemble the data to make it look more like it was originated from the original target storage device attached to a target Fibre Channel line card port. In step 92,

EX-1005 (“*Chiou*”) at 12:52-13:40.

the cache manager responsible for the target device. In step 98, the cache manager may check for available space in the SSC cache to see if, to store the current requested data, the cache manager needs to purge old cache data using either a typical least recently used cache data replacement algorithm or alternatively using pre-set time limits depending on their data types. Following step 98, the cache manager has space

Chiou at 12:52-13:40.

“using timing techniques such as *Chiou*’s (e.g., TTL) and determining whether content is valid based on a time limit would have been a simple implementation that ***reduces the need for content management*** across Glines’s system”

Mowry ¶¶ 511-513.

Pet. 33-37, 61-68; *Chiou* at 11:33-51, 12:52-13:40, Fig. 4; Mowry ¶¶ 499-561.